

Protocol for Mail / DNS

Fabian Grandke

June 5, 2010

1 Introduction

A Domain Name System (DNS) is a hierarchical naming system for computers, services, or any resource connected to the Internet or a private network. Basically it works like a phone book: it translates names to numbers and vice versa. So you may have an entry like `lkajan 192.168.16.7` in your appropriate DNS file. That brings you in the comfortable situation, that you do not have to remember the IP address, you just can use the intuitive name `lkajan`. So if you want to connect to Laszlo's computer, you just have to type his computer's name. This is very helpful because it is much easier for humans to remember names instead of numbers, especially if you can choose the names for your DNS server just 'yourself'.

A mail server is a server that manages electronic mail traffic. It handles receiving, sending, saving and forwarding actions.

An Internet Message Access Protocol (IMAP) server is a server that provides the usage of the IMAP protocol. IMAP allows the access and management from mail clients to a (remote) mail server. IMAP treats mails like they would be in a local directory. Even if you read them, the mails are located on the mail server, so you can access them from an other computer. That is also a big advantage if more than one client has access to a directory on the mail server, because the data are synchronized and are on the same level.

Icedove is a mail client we use in the course. It works pretty intuitively and has the same basic functions like other mail clients.

For the exercises it is necessary to have the following packages installed to your system:

PACKAGE NAME	PACKAGE DESCRIPTION
<i>bind9</i>	The Berkeley Internet Name Domain (BIND) implements an Internet domain name server. BIND is the most widely-used name server software on the Internet
<i>dnsutils</i>	This package delivers various client programs related to DNS that are derived from the BIND source tree.
<i>postfix</i>	Postfix is a mail transport agent that attempts to be fast, easy to administer, and secure.
<i>bsd-mailx</i>	mailx is the traditional command-line-mode mail user agent.
<i>dovecot-imapd</i>	Dovecot is a mail server whose major goals are security and extreme reliability. It tries very hard to handle all error conditions and verify that all data is valid, making it nearly impossible to crash. It should also be pretty fast, extensible, and portable. This package contains the dovecot IMAP server.
<i>icedove</i>	Icedove is a free/unbranded thunderbird mail client.
<i>procmail</i>	Can be used to create mail-servers, mailing lists, sort your incoming mail into separate folders/files, preprocess your mail, start any programs upon mail arrival or selectively forward certain incoming mail automatically to someone.

2 DNS

First of all the */etc/bind/named.conf.local* file has to be edited, so that it contains the relevant information about the mapping files. Notice that the IP address is reversed in the second zone line!

We added:

```
zone "course" {
type master;
file "/etc/bind/db.course";
};

zone "16.168.192.in-addr.arpa" {
type master;
file "/etc/bind/db.192.168.16";
};
```

In the next step the two files had to be edited themselves: In the */etc/bind/db.course* file we added:

```
;
; file that contains mapping from names to IP-addresses
;
```

```
$TTL 86400
```

```
@      IN      SOA      <your hostname>.course. root.<your hostname>.course. (
                                10051701          ; Serial
                                604800           ; Refresh
                                86400            ; Retry
                                2419200          ; Expire
                                86400 )          ; Negative Cache TTL
;
@      IN      NS       <your hostname>.course.
lkajan A        192.168.16.7
<other course members>
```

In that file the names are mapped to the static IP addresses. (e.g. lkajan to 192.168.16.7) The other members of the course had to be linked to their IP addresses also (they can be found in the wiki.

In the */etc/bind/db.192.168.16* file we added:

```
;
; file that contains mapping from IP-addresses to name
;
$TTL 86400
```

```
@      IN      SOA      <your hostname>.course.  root.<your hostname>.course. (
                                10051701          ; Serial
                                604800           ; Refresh
                                86400            ; Retry
                                2419200          ; Expire
                                86400 )          ; Negative Cache TTL
;
@      IN      NS       <your hostname>.course.
7      PTR     lkajan.course.
```

```
<other course members>
```

In that file the mapping works vice versa. The IP addresses are mapped to the names. The major part of the address is given in the */etc/bind/named.conf.local* so e.g. the 7 is just concatenated.

In the next step the command *named-checkconf -z* has to be called, to check if the file editing was valid. Then the server has to be restarted by the command */etc/init.d/bind9 restart*.

Then the file */etc/resolv.conf* has to be edited by adding the line:

```
nameserver 127.0.0.1
```

Finally you can test your name server configuration by checking the output of the following commands:

host <your username>.course

dig <your username>.course

ping <your username>.course

3 Mail

To configure your mail server you have to edit the file */etc/postfix/main.cf* by adding your static IP-address to the line **mynetworks** line. Then the server has to be restarted by the command */etc/init.d/postfix restart*. You can proof the success of the restart operation. Therefore you have to check the log using the command *\$?*. In the next step you have to add an alias to the */etc/aliases* file by adding the lines

```
postmaster:    root
root:          <your username>
```

Then the alias database has to be rebuild by the command *newaliases*. In the file *~/procmailrc* you can configure your preferred mail directory by editing the line

```
DEFAULT=' '$HOME/Maildir/''
```

To test your configuration you can send a mail to yourself by using the mail command. Now you can check your previously configured mail directory for the message.

The procmail agent uses recipes, to determine where to deliver the various mail messages. Each recipe that procmail uses consists of mode, conditions and action. Recipes are read from top to bottom. The first delivering recipe terminates the delivery process. The mode line of a recipe starts with a “:” and determines what the recipes does (e.g. :OH means “filter the header of the mail”). If the condition line exists, it determines under what conditions the action works (usually it is a regular expression). The action line determines where to save the mail (e.g. SPAM directory).

The Maildir e-mail format is a common way of storing e-mail messages, where each message is kept in a separate file with a unique name, and each folder is a directory. The local filesystem handles file locking as messages are added, moved and deleted.

As opposed to this mbox is a generic term for a family of related file formats used for holding collections of electronic mail messages. All messages in an mbox mailbox are concatenated and stored as plain text in a single file. The beginning of each message is indicated by a line whose first five characters consist of “From” followed by a space and the return path e-mail address. A blank

line is appended to the end of each message.

4 IMAP

The IMAP specific configuration is made in the file */etc/dovecot/dovecot.conf* . There are no changes to be done by the user.

Pluggable authentication modules (PAM) is a mechanism to integrate multiple low-level authentication schemes into a high-level application programming interface (API). It allows programs that rely on authentication to be written independent of the underlying authentication scheme.

In our lab PAM manages the authentication, that is necessary for the secure communication to the server.

5 Icedove

The following configuration are done most easily in the graphical environment of the mail client. At first you have to create a new Email account with the Email address `<username>@<hostname>.course`. Next choose IMAP as type and `<hostname>.course` as name of the incoming server you are using. Use the same name for your outgoing server. The last configuration step is to configure LDAP by creating a LDAP directory server with the following values:

```
Hostname:      localhost
Base DN:       dc=course
Port n:        389
Bind DN:       uid= <your username >,ou=people,dc=course
```

After configuration in Icedove you have to make sure, that your LDAP configuration serves connections to `ldap://localhost/` . That can be checked in the file */etc/default/slaped* in the line `SLAPD_SERVICES=...`

To check the Icedove and the LDAP configuration you can try to send a mail to another course member, by sending a mail from Icedove.